IJCSE International Journal of Computer Sciences and Engineering Open Access

Survey Paper Vol.-7, Issue-5, May 2019 E-ISSN: 2347-2693

A Survey on Resource Scheduling and Its Applications in Grid Environment

P. Pradeep Kumar[1*] , E. Chandra Blessie[2]

[1]Department of Computer Applications, Nehru College of Management, Coimbatore, India
[2]Department of Computer Applications, Nehru College of Management, Coimbatore, India

DOI: https://doi.org/10.26438/ijcse/v7i5.9991003 | Available online at: www.ijcseonline.org

Accepted: 20/May/2019, Published: 31/May/2019

Abstract— Grid computing is a form of distributed computing which is used to solve the large-scale scientific problems present in the grid networks. It is the technology of dividing computer networks with different and heterogeneous resources based on distributed computing. Generally, a grid network can be considered as a chain of several big branches, different kinds of microprocessors, thousands of Personal computers and workstations all over the globe. The main aim of grid computing is to apply available computing resources for dense calculations via sites that are distributed geographically without difficulty. Resource scheduling and Resource management play a key role in achieving high utilization of resources in grid computing environments. The allocation of distributed computational resources to user applications is one of the most complicated and difficult tasks in the Grid system. The problem of allocating resources in Grid scheduling requires the definition of a model that allows local and external schedulers to communicate in order to achieve efficient management of the resources themselves. This paper presents a survey of some of the most widely known and recently proposed mechanisms in Grid scheduling algorithms.

Keywords— *Grid Computing, Scheduling, Resource allocation*

I. Introduction

Grid computing is a form of distributed computing that involves coordinating and sharing computational power, data storage and network resources across dynamic and geographically dispersed organizations. Scheduling onto the Grid is NP-complete, so there is no best scheduling algorithm for all grid-computing systems. An alternative is to select an appropriate scheduling algorithm to use in a given grid environment because of the characteristics of the tasks, machines and network connectivity. Job and resource scheduling are one of the key research areas in grid computing. The goal of scheduling is to achieve the highest possible system throughput and to match the application need with the available computing resources. There are different stages of grid scheduling. The first stage is resource discovery, which provides a list of available resources. The second stage is resource allocation, which involves the selection of possible resources and the mapping of tasks to the resources. This resource allocation stage is an NP-Complete problem. The final stage includes task execution. Many heuristic algorithms have been devised to solve the task scheduling problem. Two main metrics for measuring the efficiency of task scheduling are makespan and resource utilization. Makespan is the time when grid computing system completes its latest task and an efficient scheduling algorithm is one which minimizes the idle processing time and maximizes resource utilization. This paper is organized as follows: Section 2 presents the related works of grid scheduling models. Section 3 describes the applications of resource allocation, and the conclusion and future work are discussed in section 4.

II. Related Work

A new task-scheduling algorithm named as an Efficient Task Scheduling Algorithm for Grid Computing (ETS) has been proposed by Vaaheedha Kfatheen, and Marimuthu, 2017. It is found to manage load among the available resources and result in better schedules than the compared algorithm. Their proposed algorithm has been tested using GridSim Simulation toolkit and various sets of outcomes showed that their proposed ETS algorithm produces the minimum makespan and the load on the resources is optimally balanced. The algorithm starts by identifying the tasks of minimum execution time and the respective grid resources in order to assign the task with minimum execution time on to the resource to execute first. On considering the ETC values, if the number of the lower tasks is more than the number of the higher tasks then this schedule may not be suitable to schedule them appropriately and therefore the makespan may result in a relatively large value. Then makespan, Completion Time of all the resources, Average Resource Utilization, Mean square deviation, Maximum Loaded Resource, and Tasks assigned to Maximum Loaded Resource are identified. Load balancing is an important factor to be focused seriously as it keeps all the resources active; which means that all the resources should get utilized and hence task execution would be done in a parallel fashion. When parallelism is found more than, it can be said that the execution of tasks is done at a faster rate. Scheduling heuristics like Min-Min and Max-Min schedule different tasks to different resources efficiently and it is found that load balancing among the resources is not maintained properly. Therefore, it is focused to incorporate load balancing in ETS algorithm. Experimental results obtained by their proposed algorithm for various problem sets show that it outperforms the existing scheduling algorithms.

Murugesan, and Chellappan, 2010 introduced a new resource allocation model with multiple loads of originating processors as an economic model. They found Solutions for an optimal allocation of a fraction of loads to nodes obtained to minimize the cost of the grid users via linear programming approach. The objective is to minimize the total cost of the grid user those who are assigning the job to the grid system. They assumed that the Grid system consists of five processors i.e. resources namely P1, P2, P3, P4, and P5 with four sources S1, S2, and S3 are trying to utilize the grid system to execute their workloads. After processing it is clear that the total workload of S1 is divided into three parts, the total workload of S2 is divided into three parts and the total workload of S3 is divided into four parts and allotted into processors. In addition, it shows that the completion time of each source's workloads. It is found that the resource allocation model can efficiently and effectively allocate workloads to proper resources. Experimental results showed that the proposed model obtained a better solution in terms of cost and time. They consider the problem of scheduling large-volume loads (divisible loads) within a cluster system, which is part of grid infrastructure.

Massimiliano Caramia, Stefano Giordani, 2008 proposed a tender/contract-net model for Grid resource allocation, showing the interactions among the involved actors. In the tender/contract-net protocol, Grid Resource Brokers (managers) announce their task requirements and invite bids from Grid Service Providers (contractors). Interested GSPs evaluate the requirements and submit their bids. Each GRB awards the contract to the most appropriate GSP (maximizing its utility function). They assume that the computational resource of each machine of a cluster can be split and allocated to different tasks. Task requests generated by the users are analyzed by a set of external schedulers; they assume that each external scheduler are associated with a subset of users, and are responsible only for their task submissions. The performance of their proposed market-based approach is experimentally compared with a round-robin allocation protocol, showing how the former is able to produce more effective results in terms of both system load and execution cost. In the round-robin protocol, incoming task queries are matched with the next available resource offer, which meets the task requirements and an iterator are used which cycles through the list of resource offers issued by the clusters.

Shashi Bhushan Semwal, Amit Das presented the effective time and cost-scheduling technique followed by the scheduler, decides the Grid system throughput and consumption of the source in to the grid. Their algorithm accepted the makespan and productive cost of the scheduler and minimizes the needs of processors. The schedule produced by proposed algorithm is improved than other joined bi-criteria algorithms in appreciation of together execution time and cost-effective. The Grid system is dependable for the implementation of task submits to it. The superior Grid system will contain a job scheduler which mechanically finds the most suitable machines on which a specified job is to run. This source range is very significant in dropping the total implementation time and cost of processing the jobs which depends on the job scheduling algorithm.

III. Scheduling in Grid Environment

There are two types of scheduling levels, which are involved in Grid environment one at local resource level and another at application level.

- **Resource Scheduling**

Time-sharing and Space Sharing are two types of scheduling involved at local resource level. Time-sharing scheduling is used by each system (computer) of the cluster, for example, CPU scheduler of the Operating System (OS) on each machine that is part of batch system, whereas Space Sharing scheduling is used by Local Resource Manager to schedule the job present in batch queue on idle machine of the cluster. Resource scheduling is used to increase utilization of resources or balance load on the resources. It involves taking two decisions: job selection from a queue/bag of jobs and node selection for selected job from available nodes.

- **Application Scheduling**

Various terms are used for the term Grid scheduler. These terms include super-scheduler, meta-scheduler, global scheduler, application broker, and application scheduler. Important difference between resource scheduling and application scheduling is that application level scheduler assigns a Grid site to a job whereas resource scheduler allocates a machine of a cluster to a job. Based on time at which scheduling decision is taken, scheduling can be categorized into two: Static scheduling and Dynamic scheduling.

- **Static scheduling**: It determines schedule of all the tasks of application before application starts to run. It is applicable to both independent tasks and dependent tasks applications.
- **Dynamic scheduling**: It determines schedule of a task of an application only when the task becomes ready for execution. For scheduling of independent tasks application, all tasks can be scheduled at the same time and in any order; however, for scheduling of dependent tasks application, a task becomes ready only when all its predecessor tasks have completed their execution.

Resource Allocation Mechanisms

Resource Allocation mechanisms play a significant role in allocating the most proper resources for applications. The mechanisms perform the allocation of tasks to the resources in order to make sure QoS to the application according to the user requirements. Sometimes it adopts dynamicity whereby resources are allocated as soon as they are discovered. Such mechanisms are called dynamic Resource Allocation mechanisms and are considered more efficient than static ones. Another postulation is that Resource Allocation mechanisms should be designed in such a way to avoid under utilization of resources.

Resource Allocation mechanisms provide two basic Grid services:

- Resource Monitoring
- Resource Scheduling

Resource monitoring regularly monitors resource performance, usage, capability, and future reservations. These resources include processors, memories, disks, and channel bandwidths. The information is then retrieved by the scheduler that decides on the allocation of the application to the primary resources.
The Grid Resource Allocation process is consists of four main functions:

- Resource Scheduling
- Code transfer
- Data transmission
- Resource Monitoring

Resource scheduling is an application-to-resource mapping process consisting of three main phases such as

- ➢ **Resource discovery**, where available resources are searched and a list is generated.
- ➢ **Resource selection**, which selects the best matching resource based on QoS criteria from a list of available resources generated in the resource discovery phase.
- ➢ **Job execution**, that involves submitting jobs to the selected resource(s) and monitoring their execution.

Code transfer involves transferring the code of individual tasks to the allocated resource for execution.

Data transmission concerns with data transfer needed by a task for its execution. The execution process takes place after the completion of all transfers.

Resource Monitoring step is responsible for continuous checking of resource availability, capability, usage, and future reservations.

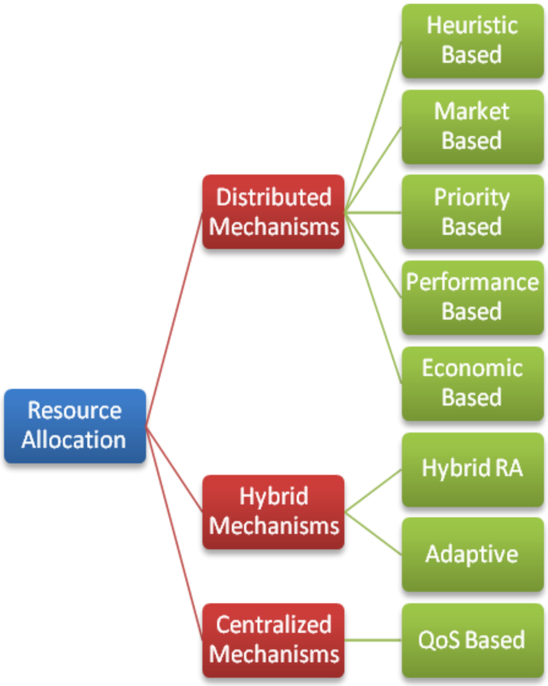

Figure 1: Resource Allocation Methods

Above figure denotes various resource methods present in the grid computing system. In that, Distributed mechanisms having a variety of scheduling methods to represent it and among them Heuristics and economic-based methods are popular ones.

Basic Scheduling Terminologies present in literature are

- ➢ **Release-time**: The initial time at which a task can start its processing in the grid.
- ➢ **Processing-time**: The time-duration needed by a task to finish its processing.

- **Start-time**: The actual time at which a task starts its processing.
- **Finish-time**: The actual time at which a task finishes its processing.
- **Expected execution time**: It is a time expected to be taken up by a task to finish its execution and is derived empirically.

Heuristics

The heuristic-based approach is to develop a scheduling algorithm, which fit only a particular type of problem, while the meta-heuristic based approach is to develop an algorithm based on a meta-heuristic method that provides a general solution method for developing a specific heuristic to fit a particular kind of problem. In general, there are five types of scheduling heuristics for different applications are given below.

- Individual task scheduling
- Batch scheduling
- List scheduling
- Cluster/Grouping based scheduling
- Duplication based scheduling

Meta-heuristics

Meta-heuristics provide both a general structure and strategy guidelines for developing a heuristic for solving computational issues. They are generally applied to a large and complex problem and it provide an efficient way of moving quickly toward a very good solution. Finally, through the above applications and surveys came to conclude that resource scheduling in distributed computing is chosen as a research area and further research is carried out in this application area.

IV. Conclusion

In this paper, various resource-scheduling and task scheduling algorithms in grid computing have been surveyed and highlighted some mechanisms, which are utilized in Grid Computing in order to schedule the resources. A comparison of various parameters like distributed, centralized, response time, hierarchical, monitoring, load balancing, resource utilization was prepared to get feedback on different types of tasks and resource scheduling. Firstly, the findings about Grid Computing and further how it works are given briefly. Then, the concept of resource allocation based on certain points of view in the literature is illustrated. The literature reviewed some of the most widely known and recently used mechanisms in task and resource scheduling. Scheduling in Grid is involved at two levels: individual site level i.e. resource scheduling and application broker level i.e. application scheduling for which this paper provided the classification of various resource scheduling algorithms and application scheduling algorithms. Further, future work will be based on the above findings to develop a more efficient algorithm for job scheduling and resource selection that will reduce the preprocessing time of jobs and considering memory constraint for resource selection.

References

[1]. S. Vaaheedha Kfatheen, A. Marimuthu, “ETS: An Efficient Task Scheduling Algorithm for Grid Computing”, Advances in Computational Sciences and Technology, Volume 10, Number 10, 2017, pp. 2911-2925.

[2]. G. Murugesan, C. Chellappan, “An Economic-based Resource Management and Scheduling for Grid Computing Applications”, IJCSI International Journal of Computer Science Issues, Volume 7, Issue 2, 2010, pp. 20-25.

[3]. Farhad Soleimanian Gharehchopogh, Majid Ahadi, Isa Maleki, Ramin Habibpour, and Amin Kamalinia, “Analysis of Scheduling Algorithms in Grid Computing Environment”, International Journal of Innovation and Applied Studies, Volume 4, Issue 3, 2013, pp. 560-567.

[4]. Raksha Sharma, Vishnu Kant Soni, Manoj Kumar Mishra, Prachet Bhuyan, “A Survey of Job Scheduling and Resource Management in Grid Computing”, World Academy of Science, Engineering and Technology International Journal of Computer and Information Engineering, Volume 4, Issue 4, 2010, pp. 736-741.

[5]. Massimiliano Caramia, Stefano Giordani, “Resource Allocation in Grid Computing: An Economic Model”, WSEAS Transactions on Computer Research, Issue 1, Volume 3, 2008, pp. 19-27.

[6]. Omar Dakkak, Suki Arif and Shahrudin Awang Nor, “Resource Allocation Mechanisms in Computational Grid: A Survey”, ARPN Journal of Engineering and Applied Sciences, Volume 10, Issue 15, 2015, pp. 6662-6667.

[7]. Ritu Garg, Awadhesh Kumar Singh, “Adaptive workflow scheduling in grid computing based on dynamic resource availability”, Engineering Science and Technology, an International Journal, 2015, pp. 1-14.

[8]. Boopathi Kumar E, and Thiagarasu V, “Big Data and Its Applications: A Review”, International Journal of Intelligent Computing and Technology (IJICT), Volume 2, Issue 2, 2019, Pages: 1 – 10.

[9]. Sonal Nagariya, Mahindra Mishra, “Resource Scheduling in Grid Computing: A Survey”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 10, 2013, pp. 735-739.

[10]. Shiv Prakash, Deo Prakash Vidyarthi, “A Hybrid GABFO Scheduling for Optimal Makespan in Computational Grid”, International Journal of Applied Evolutionary Computation, Volume 5, Issue 3, 2014, pp. 57-83.

[11]. Harshadkumar B. Prajapati, Vipul A. Shah, “Scheduling in Grid Computing Environment”, Fourth International Conference on Advanced Computing & Communication Technologies, IEEE, 2014, pp. 315-324.

[12]. Muhammad Bilal Qureshi, Maryam Mehri Dehnavi, Nasro Min-Allah, Muhammad Shuaib Qureshi, Hameed Hussain, Ilias Rentifis, Nikos Tziritas, Thanasis Loukopoulos, Samee U. Khan,

Cheng-Zhong Xu. Albert Y. Zomaya, "Survey on Grid Resource Allocation Mechanisms", Journal of Grid Computing, Springer, 2014, pp. 1-43.

[13]. Zahra Pooranian, Mohammad Shojafar, Jemal H. Abawajy, Ajith Abraham, "An efficient meta-heuristic algorithm for grid computing", Journal of Comb Optim, Springer, 2013, pp.1-22.

[14]. Shashi Bhushan Semwal, Amit Das, "Effective Time and Cost Based Task Scheduling In Grid Computing", International Journal of Computer Science Trends and Technology (IJCST) – Volume 3 Issue 5, Sep-Oct 2015, pp. 118-122.

Authors Profile

Mr. C T Lin pursed Bachelor of Science from University of Taiwan, Taiwan in 2006 and Master of Science from Osmania University in year 2009. He is currently pursuing Ph.D. and currently working as Assistant Professor in Department of Computational Sciences, Department of Electronic and Communication, University of Taiwan, Taiwan since 2012. He is a member of IEEE & IEEE computer society since 2013, a life member of the ISROSET since 2013, ACM since 2011. He has published more than 20 research papers in reputed international journals including Thomson Reuters (SCI & Web of Science) and conferences including IEEE and it's also available online. His main research work focuses on Cryptography Algorithms, Network Security, Cloud Security and Privacy, Big Data Analytics, Data Mining, IoT and Computational Intelligence based education. He has 5 years of teaching experience and 4 years of Research Experience.

Mr C H Lin pursed Bachelor of Science and Master of Science from University of New York, USA in year 2009. He is currently pursuing Ph.D. and currently working as Assistant Professor in Department of Telecommunication, University of New York, USA since 2012. He is a member of IEEE & IEEE computer society since 2013, a life member of the ISROSET since 2013 and ACM since 2011. He has published more than 20 research papers in reputed international journals including Thomson Reuters (SCI & Web of Science) and conferences including IEEE and it's also available online. His main research work focuses on Cryptography Algorithms, Network Security, Cloud Security and Privacy, Big Data Analytics, Data Mining, IoT and Computational Intelligence based education. He has 5 years of teaching experience and 4 years of Research Experience.